RECENT IMPROVEMENTS TO IBM'S SPEECH RECOGNITION SYSTEM FOR AUTOMATIC TRANSCRIPTION OF BROADCAST NEWS

S. S. Chen, E. M. Eide, M. J. F. Gales, R. A. Gopinath, D. Kanevsky, P. Olsen
IBM Thomas J. Watson Research Center

ABSTRACT

We describe recent improvements to IBM's system for automatic transcription of broadcast news. Some of the improvements are: Bayesian Information Criteria (BIC) applied to choosing the number of components in a Gaussian mixture model, tail distribution modelling using Richter distributions and power exponential distributions, pronunciation networks, adaptive training techniques such as clustered adaptive training (CAT) and a modified version of speaker adaptive training (SAT) which is efficient for large tasks, factor analysis invariant to linear transforms. We also experimented with changes such as changing the acoustic vocabulary, reducing the number of phonemes and insertion of short pauses. The models were combined in a single system using NIST's script voting machine known as ROVER.

1. INTRODUCTION

Recently the focus of research in large vocabulary continuous speech recognition recognition (LVCSR) has been shifted from read speech data to speech data found in the real world - like broadcast news over radio and TV and conversational speech over the telephone. A considerable amount of both acoustic (approximately 200 hours of which about 80% is usable) and linguistic (approximately 400 million words) training data for broadcast news has been made available by the Linguistic Data Consortium (LDC) in the context of DARPA sponsored Hub4 evaluations of large vocabulary continuous speech recognition (LVCSR) systems on broadcast news [14]. Broadcast news transcription poses several challenges to LVCSR systems. First, automatic segmentation of the input audio stream is required. Second, the speech data exhibits a wide variety of speaking styles, environmental and background noise conditions and channel conditions. These are categorized as the so-called F-conditions [14]: prepared speech (F0), spontaneous speech (F1), low fidelity speech, including telephone channel speech (F2), speech in the presence of background music (F3), speech in the presence of background noise (F4), speech from non-native speakers (F5) and FX - all other speech.

The LVCSR systems participated in the broadcast news transcription task [14] can be categorized as two types. The first type of systems are conglomerate: the entire training data which consists of various types of speech were pooled together to train one single acoustic model. The second type of systems are condition dependent: different acoustic models were built for different conditions, e.g. gender dependent acoustic models and F-condition dependent acoustic models.

The IBM LVCSR system used in the 1997 evaluation was a conglomerate system which had 3.5K HMM states and 170K Gaussians. It was trained in the optimal feature space [8, 11] using 80 hours of acoustic training data provided by LDC. The decision trees for the HMM states were built using the relatively clean data from the F0 and F1 conditions, whereas the Gaussian mixtures were trained on the complete set of training data. We also designed a successful automatic segmentation and clustering algorithm which is based on the Bayesian information criterion [5]. After baseline decoding, we performed iterative MLLR unsupervised adaptation on both means and variances [9] for each cluster.

In this paper we present algorithmic improvements we have made this year. Some of the highlights are: Bayesian Information Criteria (BIC) applied to choosing the number of components in a Gaussian mixture model, tail distribution modelling using Richter distributions and power exponential distributions, pronunciation networks, adaptive training techniques such as clustered adaptive training (CAT) and a modified version of speaker adaptive training (SAT) which is efficient for large tasks, factor analysis invariant to linear transforms. We will also describe the IBM system used in the 1998 evaluation which consists of multiple systems combined by the NIST's script voting program known as ROVER [7].

2. OVERVIEW OF THE LVCSR SYSTEM

The IBM LVCSR system uses acoustic models for sub-phonetic units with context-dependent tying (see [1, 2] for details). The instances of context dependent sub-phone classes are identified by growing a decision tree from the available training data [1] and specifying the terminal nodes of the tree as the relevant instances of these classes. The acoustic feature vectors that characterize the training data at the leaves are modeled by a mixture of Gaussian or Gaussian–like pdf's, with diagonal covariance matrices. The HMM used to model each leaf is a simple 1-state model, with a self-loop and a forward transition.

In addition to the 80 hours of acoustic training data we had in 1997, LDC provided 80 hours of extra data this year, however with no annotation on the F conditions. We decided to use the full set of data to build decision trees containing a total of 3.5K HMM states. The Gaussian mixtures were built from the full training data and the best single system we arrived at contained 289K Gaussian. The technique for finding optimal feature spaces developed last year was used in all models used in our current system.

3. ALGORITHMIC IMPROVEMENTS

For all our development work, we used the 1997 hub4 evaluation set with the hand segmentation provided by NIST. Table 1 displays the error rate using our 1997 evaluation baseline system. To speed up our work, we subsampled the test set in some experiments.

3.1. Bayesian Information Criterion

One problem in Gaussian mixture modelling is how to choose the number of Gaussians. It is well-known that too few Gaussians does not give sufficient model complexity whereas too many leads to overtraining. Our goal here is to adaptively choose the number of Gaussians according to the underlying complexity of the HMM state.

A common heuristic solution of this problem is the *thresholding* method. According to the number of samples belonging to the HMM state in the training data, one choose the number of Gaussians proportionally.

	All	F0	F1	F2	F3	F4	F5	FX
1997 Base	19.8	12.1	19.6	29.9	25.8	25.9	22.3	38.7

Table 1. 1997 Baseline system on the 1997 evaluation full set.

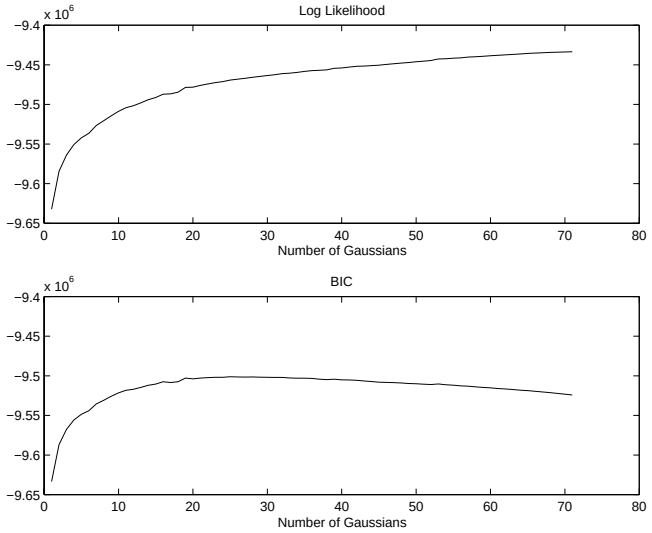

Figure 1. Choosing the number of Gaussians by maximizing the BIC criterion

In this paper, we propose to choose the number of Gaussians by optimizing the Bayesian Information Criterion (BIC), a well-known model selection criterion in the statistics literature. For a particular HMM state, let n be the number of mixture components, C_n the clustering corresponding to n mixtures, N_{C_n} the number of parameters used in the mixture and N the sample size. We define the BIC function BIC(n) as follows

$$\mathrm{BIC}(n) = \log(\mathrm{Likelihood}(C_n)) - \frac{\lambda}{2} * N_{C_n} * log(N)\ . \quad (1)$$

We choose n by maximizing the BIC function:

$$\hat{n} = \arg\max BIC(n).$$

Figure 3.1. illustrates how this procedure works for a particular HMM state. The horizontal axis represents the number of Gaussians. The vertical axis represents the log-likelihood in Panel (a) and the BIC value in Panel (b). Clearly as the number of Gaussians increases, the likelihood always improves, whereas the BIC value first increases then declines. The BIC value is optimized at $n = 27$.

We conducted experiments comparing the BIC approach with the heuristic thresholding method. We designed a system by the thresholding method which had 90K Gaussians. By choosing the penalty weight $\lambda = 1$, we obtained a system which had roughly 90K Gaussians using the BIC method. Figure 3.1. plots each HMM state by the its training sample size and its number of Gaussians determined by the BIC procedure. Notice that a certain state belonging to `F-2` and a certain state belonging to `AO-2` are indicated in the figure. They both had roughly the same number of samples. It is interesting that the BIC procedure chose about 25 Gaussians for the state belonging to `F-2` whereas about 105 Gaussians for the state belonging to `AO-2`. In fact, we found out that most of the "upper" states, which have big angles from the horizontal axis if connected with the origin, are mostly vowels; most of the "lower" states, which have small angles from the horizontal axis if connected with the origin, are most consonants. This shows that the BIC procedure indeed tends to choose more Gaussians for more complex states. Table 2 shows that the system built by the BIC procedure outperformed the system built by the thresholding method, by 0.8% absolute. In fact, in our experiments, we have observed consistently that compared

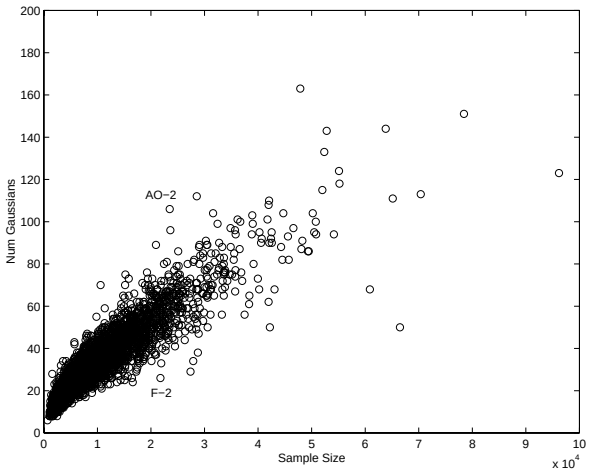

Figure 2. The BIC procedure tends to assign more Gaussians to more complex states

with the thresholding method, the BIC approach can produce systems achieving reduced error rate with the same number of Gaussians, or produce systems achieving the same error rate but with smaller number of Gaussians.

By varying the penalty weight λ in the BIC criterion, we can obtain systems with various numbers of Gaussians. As we decreased λ, we obtained systems with increasing numbers of Gaussians. As indicated in Table 2, the recognition accuracy dropped. We decided to use the 289K system as our baseline.

3.2. Power Exponential Distributions

When viewing histograms of 1–dimensional projections of the acoustic feature vectors in the training data, one is struck by the sharpness and asymmetries of the peaks of the histograms. It is rather difficult to capture these features using Gaussian models. Instead, we propose to use multidimensional generalizations of the power exponential distribution, which is also known as the alpha stable distribution. For feature vector $x \in R^d$, the power exponential density with power α, mean μ and diagonal covariance σ is the following:

$$f(x;\mu,\sigma,\alpha) = \rho_\alpha \exp\left\{ -\left(\gamma_\alpha \sum_{j=1}^{d} \frac{(x_j-\mu_j)^2}{2(\sigma_j)^2} \right)^{\frac{\alpha}{2}} \right\}, \quad (2)$$

where rho_α and γ_α and normalizing constants which depends only on α. Notice that γ_α is necessary so that $sigma$ means the covariances.

When $\alpha = 2$, the power exponential distribution (2) is exactly the multivariate Gaussian distribution with diagonal covariances. When $\alpha = 1$, (2) is the Laplacian density, which is used in the Phillips systems [13]. When $\alpha < 2$, the power exponential density (2) is sharper at origin and has slower decaying tails than the Gaussian density; when $\alpha < 2$, the opposite holds.

We can build acoustic models using mixtures of power exponential distributions. If α is fixed at a certain value for all HMM states, the maximum likelihood solution of the parameters (means, variances, mixture weights) can be obtained approximately via an EM algorithm. Moreover, the power α for each HMM state can be optimized individually so that each HMM state has its own $variable\ \alpha$. See [12] for the details of these algorithms.

	# Gaussians	All	F0	F1	F2	F3	F4	F5	FX
Standard	90K	26.0	11.9	23.5	31.7	28.4	28.5	22.3	42.3
$\lambda = 1.00$	90K	25.2	11.6	23.1	30.5	27.7	26.2	20.5	41.8
$\lambda = 0.80$	135	24.7	11.2	21.2	29.5	29.0	26.8	21.6	41.2
$\lambda = 0.65$	178	24.2	10.7	21.5	29.3	26.5	25.9	21.4	40.3
$\lambda = 0.54$	237	23.8	10.7	21.6	29.3	26.5	24.2	19.7	39.6
$\lambda = 0.45$	289	23.5	10.5	21.5	28.9	24.4	24.6	20.7	39.0

Table 2. Comparison of the BIC approach with the thresholding approach on the 1997 evaluation subset.

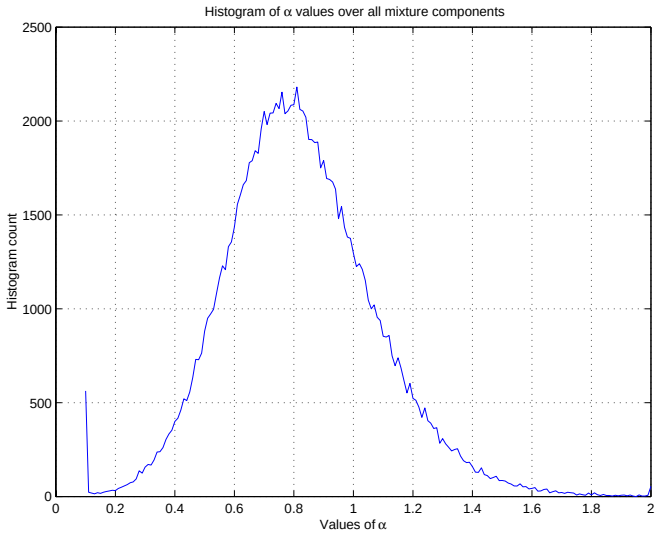

Figure 3. Histogram of the optimized α's

We built systems with $\alpha = 2$, $\alpha = 1$, and variable α's; they all had 100K components. Table 3 displays the recognition results on the 1997 evaluation subset. The system with $\alpha = 1$, which corresponds to the Laplacian distributions, outperformed the system with $\alpha = 2$, which corresponds to the standard Gaussian mixture distributions, by 0.5% absolution. The system with variable α's was only slightly better than the system with fixed $\alpha = 1$. Figure 3.2. plots the histogram of the variable α's. It is interesting that most of the optimized α's were around $\alpha = 1$! In our final α system, we actually fixed $\alpha = 1$.

Unfortunately, when we increased the number of components to 280K, the power exponential system became worse than the standard Gaussian system. This might be caused by possible numerical problems of our estimation algorithms.

3.3. Richter Distributions

To improve the modelling of distribution tails, we can also use the so-called Richter distribution which was proposed by Allen Richter a decade ago [15, 3]. A Richter distribution is a mixture of Gaussians which are centered at the same mean with scaled version of the same covariances:

$$R(x; \mu, \Sigma, c, \lambda) = \sum_i \lambda_i N(x; \mu, c_i \Sigma)$$

where μ is the mean, Σ is the covariance matrix and c_i are the scaling factors; we refer to each Gaussian here as a Richter component. In our experiments, we replace each Gaussian in our standard Gaussian mixture system by a Richter distribution with 4 Richter components. Since the only extra parameters are the scaling factors c_i, which are very few, there is minimal computational overhead. Since ultimately there are only Gaussians in the model, they can be adapted using standard techniques such as MLLR. See [12] for details of our implementation.

The scaling factors c_i can be tied as various levels: global level, HMM state level, or Gaussian level. Table 4 shows our experiments on the Hub4 1997 evaluation full set. State level tying Richter gave improved performance (18.7%) compared with the standard Gaussian mixture system (18.4%). This indicates that the standard Gaussian mixture components are ill suited at modelling the tails of the distributions. However, gains were greatly reduced after performing mean and variance MLLR adaptation.

4. PRONUNCIATION NETWORKS

Our goal is to model actual pronunciations by modifying the acoustic model topology. Words in the speech recognizer are mapped to strings of phonemes. In the standard systems, each phoneme is expanded as the regular 3 state topology. In our approach, each phoneme is associated with a decision tree which determines the pronunciation network topology according to the phonetic context: depending on its phonetic context, a phoneme will end at a certain node of the tree which is associated with a network topology; we plug in that network topology into the HMMs. We emphasize that this decision tree is not the decision tree for determining context-dependent HMM states.

we build the decision trees and networks as follows. First, a "ballistic" decoding that decodes as if the leaves were words, is performed on the training data. The string of decoded leaves are then aligned to the "correct" labels prescribed by a training transcription so that each "correct" leaf is assigned a string of ballistic leaf labels. Pairs of leafs and ballistic leaf strings with high co–occurrence counts are selected to build a the tree and the network. This technique is an extension of work done on Fenonic modelling at IBM during the late eighties and early nineties. See [6] for the details of our approach.

The actually network topology we obtained in our experiments could be the regular 3 state topology, or with reduced pronunciation, or with inserted pronunciation, or with substituted pronunciation. In our experiment, 89% of the topologies were actually the standard 3 state topology; 4.1% were the standard 3 state topology with a skip; 1.8% had four states, etc. In recognition, the pronunciation network models appear to improve F1 (spontaneous speech) as would be expected, as indicated in table 5. In the 1998 evaluation, we applied the pronunciation network models at the last iteration of the iterative MLLR adaptation.

4.1. Adaptive Training

The clustered adaptive training (CAT) technique was presented by Gales at ICASLP 1998 [10]. In our particular implementation, for each speaker (in general for each cluster), its mean was computed as linear combination of 4 *canonical* means:

$$\mu = \sum_{i=1}^{4} \lambda_i^{(s)} \mu_i.$$

The four canonical means were initialized by applying supervised MLLR on the 289K BIC system toward the four conditions: {Clean , Noisy} × {Male, Female}. The canonical means and the shared covariance were estimated by maximum likelihood via EM. During testing, given hypothesized script, first the weights λ were estimated by maximum likelihood via EM; then the mean were adapted. CAT is suited form rapid adaptation, since estimating the weights λ requires very small amount of data.

We also implemented a modified version of SAT. In the conventional SAT originally proposed by BBN, the mean of each speaker is obtained by a linear transform on the canonical

	All	F0	F1	F2	F3	F4	F5	FX
$\alpha \equiv 2$	26.1	11.8	22.9	32.1	27.9	27.7	23.1	43.9
$\alpha \equiv 1$	25.5	11.5	23.0	31.3	28.1	27.6	21.6	41.1
variable α	25.4	11.9	22.6	31.3	29.0	26.5	21.8	41.1

Table 3. Using Mixtures of power exponential distributions on the 1997 evaluation subset.

	All	F0	F1
Base	18.7	11.6	18.5
Base+MLLR	16.4	10.1	17.0
Global Richter	18.5	11.5	18.3
State Richter	18.4	11.3	18.1
Gaussian Richter	18.5	11.5	18.2
State Richter + MLLR	16.3	10.1	16.9

Table 4. Performance of the Richter systems on the 1997 evaluation full set.

mean

$$\mu^{(s)} = A^{(s)}\mu.$$

In order to update μ in the EM iteration, one needs to store a full matrix for each Gaussian in the system. This turns out to be a bottle neck for large systems. In our modified version of SAT [9], each speaker is associated with a feature space transform:

$$x^{(s)} = A^{(s)}x,$$

which is equivalent to a constrained model space transform:

$$\mu^{(s)} = A^{(s)}\mu \quad ; \quad \Sigma^{(s)} = A^{(s)}\Sigma A^{(s)^T}.$$

Since the operation is on the feature vectors, one can update μ in the standard fashion in the EM iteration, however with respect to the transformed feature $x^{(s)}$. It is practical for large systems, and requires small changes to the standard code. In our experiments, the SAT system was initialized as the 289K BIC system.

Table 6 compares the performance of CAT, SAT and MLLR (mean + variance) [9] on the 1997 evaluation full set. SAT+MLLR outperformed MLLR along by 0.5% absolute. Compared with our 1997 evaluation results, we gained 2.3% absolute in the baseline by extra training data and by choose number of Gaussians via BIC, and another 0.5% absolution on adaptation by applying SAT.

4.2. Factor Analyzed Covariances

Let j be an index referring to a specific mixture component. To better model covariances without modelling the full covariance matrices Σ_j, we constrain the covariances to be of the form $\Sigma_j = A(\Lambda_j\Lambda_j^T + \Psi_j)A^T$ where A is a shared matrix capturing an optimal feature space, Λ_j is a "factor loading matrix" whose columns are less abundant than those of Σ_j, typically numbering 2 or 3 columns, and Ψ_j is a diagonal specific matrix. Methods for parameter estimation of Gaussian mixtures with covariances of this form are described in [8] and the method is named factor analyzed covariances invariant to linear transformations (FACILT). Some initial experiments with 2 column factor loading matrices are shown in Table 7. The only condition that improved significantly was FX. Experiments with different number of factors and tying structures of the covariances are still ongoing.

4.3. Short Pause

Previously our silence phone consisted of a 3-state Hidden Markov Model. This we felt was insufficient for modelling short pauses. To address this problem a new deletable short pause phone `SX` was introduced at the end of each word. `SX` is modelled by a single deletable one-state Hidden Markov Model. This phone was introduced into our system and models retrained with the new phone. The idea being that short silences would not be "eaten up" by other phones at the endings and beginnings of words. The short pause appears to improve the conditions F0, F1 and FX as can be seen in Table 8

4.4. Pronunciation Dictionary

As our phonetic spellings, also known as baseforms, have been added to and composed in many different ways, the current list of baseforms comes from a variety of sources and contains many inconsistencies. To remove these inconsistencies we inspected spellings of words with common prefixes and suffixes. In addition we allowed words like "Human" with baseform `HH Y UW M AX N` to delete the `HH` as is done in some dialects of American–English. In baseforms where `Y UW` was preceded by a dental (`T, D, TH` or `D`) (e.g. as in duty `D Y UW T IY` or `D UW T IY`) we allowed the `Y` to be deleted for a similar reason. Lastly we went through words ending in "ING" and compared the baseforms to the baseform of it's root. The list of baseforms produced in this fashion was dubbed "clean". The resulting vocabulary gave little improvements. A comparison is shown in Table 9.

4.5. The Phone Set

We deleted 10 phones that we felt were treated erroneously and/or inconsistently in our set of baseform. These phones were `AXR, AH, BD, DD, GD, IH, KD, PD, TD` and `TS`. `BD, DD, GD, KD, PD` and `TD` are phones that were intended to model "double stops", i.e. stops that were followed by new stops and `TS` and `AXR` to model "`T S`" and "`AX R`" that was felt were such short sounds that individual phones had to be introduced. `AH` and `IH` are sounds that are very close to already existing sounds that are not distinguished well in our baseform set. After replacing all these phones in the acoustic dictionary we trained new Gaussian models and compared with the existing phone set. The results were significantly worse, cf. Table 10, but as seen in section 5.4. it helped yield an improved system when mixed with other pre-existing systems using rover.

5. 1998 IBM SYSTEM

5.1. Segmentation

We first applied the BIC change detection scheme [5] to detect acoustic changes in the data. According to the detected changes, the entire audio stream was chopped into turns. Because some turns were quite long, we further chop each turn into smaller segments according to the silence information; also the silence information was used to prevent segment boundaries from splitting words. We performed classification to reject the pure music segments; the classification was based on Gaussian mixtures models [2]. Table 11 compares the NIST hand segmentation and our automatic segmentation on the 1998 hub4 evaluation set in terms of recognition error rates using the 289K BIC system; our automatic segmentation increased the error rate by only .3% in the first set.

5.2. Baseline Systems

Over the year, while investigating various techniques, we had a range of systems:

- 289K standard BIC system.
- 271K system using a new phone set.
- 289K telephone bandwidth system built by reducing the bandwidth of the training data to 4KHz.
- 289K power exponential system.

	All	F0	F1	F2	F3	F4	F5	FX
3 state topology	22.6	9.1	20.8	28.0	25.1	24.4	19.6	37.1
pronunciation nets	22.4	8.9	20.1	27.8	25.0	24.4	19.5	37.4

Table 5. Comparison the Pronunciation networks with traditional tristate HMM models on the 1997 evaluation subset

	All	F0	F1	F2	F3	F4	F5	FX
289K Base	17.5	10.7	17.7	26.5	23.9	21.9	17.5	34.4
MLLR	15.6	9.8	16.2	21.2	21.3	20.4	14.6	32.3
CAT + MLLR	15.2	9.5	15.5	21.3	21.3	18.9	15.2	32.1
SAT + MLLR	15.1	9.6	15.7	20.3	20.8	18.1	15.2	31.7

Table 6. Performance of adaptive training on 1997 evaluation set

- 93K left-context Only System designed for the 10 times real time task.
- 289K Alpha-Mixtures System.

The decoder in all five systems is a single-pass decoder which employs the rank-based decoding strategy and the envelope search algorithm [1]. Table 12 shows the error rates of the 5 baseline decodes. It is clear that the 5 systems performed quite differently. Among them, the 289K BIC system achieved the best error rate.

5.3. Clustering and Unsupervised Adaptation

After segmentation, the segments were clustered using a standard maximum-linkage bottom-up-clustering procedure with a single Gaussian model for each segment and log-likelihood ratio distance measure. The termination for this bottom-up-clustering procedure was determined to maximize the BIC criterion [5].

After clustering, we performed iterative MLLR on each cluster. At the first two iterations, both a mean transform and an efficient full-variance transform were estimated [9]; subsequently, only means were adapted. Totally 6 iterations of MLLR were performed. In addition, we had the CAT system and the SAT system described in section 4.1.. The CAT and SAT transforms were estimated using the BIC Base System scripts after the first 2 iterative MLLR adaptation. Once the transforms were computed, CAT and SAT followed the same iterative MLLR adaptation procedure as the baseline systems. Table 12 displays the adaptation improvements in word error rate (We have yet not implemented the adaptation scheme for the Alpha mixture system). MLLR on top of CAT and SAT performed the best; we achieved about 15% reduction in word error rate. We also notice that most of the gain in the iterative MLLR came in the first two iterations.

5.4. Rover

J. Fiscus introduced a voting scheme for combining word scripts produced by different speech recognizers, [7]. This program was named *ROVER*. We applied *ROVER* on the scripts decoded by the above 7 different systems. As indicated in Table reftable-rover, our best individual system is the SAT system, which achieved 15.5% on the first set and 12.8% on the second set. *ROVER* reduced the error rate by 5.8% relative on the first set; however the gain on the second set was very minimum (1.2% relative).

REFERENCES

[1] L. R. Bahl et al., "Robust Methods for using Context-Dependent features and models in a continuous speech recognizer", *Proc. ICASSP*, 1994.

[2] L. R. Bahl et al., "Performance of the IBM large vocabulary continuous speech recognition system on the ARPA Wall Street Journal task", *Proc. ICASSP*, pp 41-44, 1995.

[3] P. F. Brown, "The acoustic-modelling problem in Automatic speech recognition by," PhD thesis CMU, 1987.

[4] S. S. Chen et al., "IBM's LVCSR System for Transcription of Broadcast News Used in the 1997 Hub4 English Evaluation," *Proc. of DARPA Speech Recognition Workshop*, Feb 8–11, Lansdowne VA, 1998.

[5] S. Chen et al, "Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion", *Proc. of DARPA Speech Recognition Workshop*, Feb 8–11, Lansdowne VA, 1998.

[6] E. Eide, "Automatic Modeling of Pronunciation Variations", else where in this proceedings.

[7] J. G. Fiscus, "A post–processing system to yield reduced word error rates: recognizer output voting error reduction (rover)," *technical report National Institute of Standards and Technology*, 1997.

[8] R. A. Gopinath, "Constrained Maximum Likelihood Modelling with Gaussian Distributions," *Proc. of DARPA Speech Recognition Workshop*, Feb 8–11, Lansdowne VA, 1998.

[9] M.J.F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition", *Computer Speech and Language*, Vol 12, p 75-98, 1998.

[10] M. J. F. Gales, "Cluster Adaptive Training for Speech Recognition", *Proc. of ICSLP, Sydney, Australia*, 1998.

[11] M. J. F. Gales, "Semi-tied covariance matrices for hidden Markov Models", *IEEE Trans. Speech Audio Processing*, Vol 7, to appear, 1999.

[12] M. J. F. Gales and P. A. Olsen, "Tail Distribution Modelling Using the Richter and Power Exponential Distributions," submitted to EuroSpeech 99.

[13] R. Haeb–Umbach, et al., "Acoustic modelling in the Phillips Hub4 continuous speech recognition system," *Proc. of DARPA Speech Recognition Workshop*, Feb 8–11, Lansdowne VA, 1998.

[14] D. Pallet, "Overview of the 1997 DARPA Speech Recognition Workshop," *Proc. of DARPA Speech Recognition Workshop*, Feb 2-5, Chantilly VA, 1997.

[15] A.G. Richter, "Modelling of continuous speech observations," *Advances in Speech Processing Conference*, IBM Europe Institute, 1986.

	All	F0	F1	F2	F3	F4	F5	FX
FACILT	22.7	9.9	20.3	27.3	26.1	24.8	19.8	37.1
Standard	22.6	9.6	20.3	27.2	25.9	23.9	19.7	38.0

Table 7. Comparison of the FACILT with a comparable diagonal Gaussian model with an equivalent number of prototypes on the 1997 evaluation subset.

	All	F0	F1	F2	F3	F4	F5	FX
no short pause	18.5	11.5	18.2	27.3	25.6	25.4	18.8	35.9
with short pause	18.3	11.4	18.3	27.1	24.6	23.4	18.4	35.5

Table 8. The effect of using short pause on the 1997 evaluation set.

	All	F0	F1	F2	F3	F4	F5	FX
Old	25.2	11.4	22.5	30.8	27.6	28.2	21.0	40.6
Clean	25.1	11.2	23.2	30.6	27.7	26.5	21.4	40.8

Table 9. Performance of the clean pronunciation vocabulary on the 1997 evaluation subset.

	All	F0	F1	F2	F3	F4	F5	FX
Old	25.2	11.4	22.5	30.8	27.6	28.2	21.0	40.6
New	27.8	13.9	25.0	33.1	31.3	30.2	26.0	43.1

Table 10. Comparison of the new phone set with old phone set on the 1997 evaluation subset.

	All	F0	F1	F2	F3	F4	F5	FX
NIST Set I	18.0	8.9	19.9	27.9	29.4	12.9	24.8	25.2
IBM Set I	18.3	8.9	19.6	28.8	28.8	13.1	22.4	26.5
NIST Set II	15.1	9.6	16.5	20.3	16.0	18.4	15.7	40.2
IBM Set II	15.1	9.5	16.2	20.3	16.4	18.0	12.9	43.3

Table 11. Segmentation on 1998 evaluation set

	289K	New Phone Set	Tele	Left	SAT	CAT
Set I: Base	18.3	19.1	22.7	20.9	18.3	18.3
Set I: MLLR	15.7	16.3	18.4	17.5	15.5	15.4
Set II: Base	15.1	16.9	20.4	17.4	15.1	15.1
Set II: MLLR	13.3	14.7	16.6	14.9	12.8	13.1

Table 12. Adaptation on 1998 evaluation set

Set I	All	F0	F1	F2	F3	F4	F5	FX
289K	15.7	8.0	18.4	22.6	25.0	10.5	21.8	21.9
New Phone Set	16.3	8.6	19.7	24.6	24.3	11.3	19.4	21.5
Tele	18.4	10.1	21.1	24.4	31.5	12.9	23.6	24.8
Left	17.5	9.7	19.7	27.1	29.2	11.7	23.6	24.0
Alpha	18.9	8.9	19.2	30.0	31.3	13.9	27.9	27.6
SAT	15.5	7.8	18.0	22.2	26.6	10.5	22.4	21.1
CAT	15.4	7.8	17.5	23.2	26.4	10.5	24.2	21.3
Rover	14.5	7.8	16.8	20.9	24.7	10.0	19.4	19.7
Set II	All	F0	F1	F2	F3	F4	F5	FX
289K	13.3	8.6	15.7	16.0	15.3	15.0	5.7	37.8
New Phone Set	14.7	10.1	18.2	14.6	16.4	16.4	14.3	37.8
Tele	16.6	10.8	17.5	17.0	16.9	19.9	5.7	50.5
Left	14.9	9.6	17.8	17.9	16.3	16.2	8.6	48.0
Alpha	16.0	10.5	18.2	20.4	16.2	17.6	21.4	50.2
SAT	12.8	8.5	14.8	14.6	13.8	14.4	5.7	37.4
CAT	13.1	8.8	15.7	16.8	14.0	14.1	5.7	38.4
Rover	12.6	8.5	14.6	14.9	13.6	14.2	5.7	34.1

Table 13. *ROVER* on 1998 evaluation set